Challenges and Governance Solutions for Data Science Services based on Open Data and APIs

Joutsenlahti, Juha-Pekka

IEEE
2021

Joutsenlahti , J-P , Lehtonen , T , Raatikainen , M , Kettunen , E & Mikkonen , T 2021 , Challenges and Governance Solutions for Data Science Services based on Open Data and APIs . in 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN) of 43rd International Conference on Software Engineering (ICSE) . IEEE , Workshop on AI Engineering - Software Engineering for AI , 30/05/2021 . https://doi.org/10.1109/WAIN52551.2021.00012

http://hdl.handle.net/10138/333646
https://doi.org/10.1109/WAIN52551.2021.00012

unspecified
acceptedVersion

Challenges and Governance Solutions for Data Science Services based on Open Data and APIs

Juha-Pekka Joutsenlahti and Timo Lehtonen
Solita Ltd, Finland
first.last@solita.fi

Mikko Raatikainen, Elina Kettunen and Tommi Mikkonen
University of Helsinki, Finland
first.last@helsinki.fi

***Abstract*—Increasingly common open data and open application programming interfaces (APIs) together with the progress of data science – such as artificial intelligence (AI) and especially machine learning (ML) – create opportunities to build novel services by combining data from different sources. In this experience report, we describe our firsthand experiences on open data and in the domain of marine traffic in Finland and Sweden and identified technological opportunities for novel services. We enumerate five challenges that we have encountered with the application of open data: relevant data, historical data, licensing, runtime quality, and API evolution. These challenges affect both business model and technical implementation. We discuss how these challenges could be alleviated by better governance practices for provided open APIs and data.**

***Index Terms*—Open data, application programming interface, API, artificial intelligence, machine learning, governance.**

I. Introduction

Especially governmental organizations and agencies provide different types of open data sources in several countries today. While not new, open data has not always been available via convenient Application Programming Interfaces (APIs) as the data can also be provided only as documents [1]. However, for example in Finland, this will be resolved because the recent Finnish legislation demands governmental organizations to provide APIs for their public data adhering to the European Union directive [2]. The same legislation, thus, increases the amount of available open data and open APIs as different data will be opened. Additionally, this enables more near real-time data when data will become available automatically through the APIs.

As the quantity of data and data sources grow massively, a need and opportunity emerge for data science services that can process huge amounts of data. Data can be seen as a service proposal per se, but with open data, the commercial product is rather the intelligence built on top of the data to solve the specific contextual problems of the customers. More and more often, this takes place in the form of artificial intelligence (AI) and machine learning (ML) systems.

However, provided APIs essentially define and control what operations can be performed on open data, by whom, and under what conditions. For open data usage, there are challenges, such as legal or privacy issues and possible changes in governmental policies [3]. Challenges are also associated with the creation and evolution of data science services, such as ensuring adequate efficiency in processing large data sets (data-intensive flows management) [4].

This experience report discusses the following problem: How to build data science services on top of a set of open data providers by combining open data sources and carrying out advanced analyses, such as machine learning so that the results are valuable for end-customers? Our firsthand experiences originate from the development of open data and APIs for marine traffic. This open data in association with other open data sources introduce new sustainable data science service opportunities for end-customers based on various AI technologies. Towards this end, we enumerate the challenges we have identified related to these opportunities and propose how API governance could alleviate the challenges.

II. Case: Artificial intelligence for the Finnish–Swedish Winter Navigation System

A. Background: Marine traffic

We base our firsthand experiences on marine traffic data in Finland and Sweden, especially during wintertime when the Finnish–Swedish Winter Navigation System (FSWNS) [5] is active. FSWNS maintains safe and efficient year-round navigation with agreements and information systems. The Finnish public authority (Fintraffic) provides near real-time monitoring of traffic through public APIs[1]. This case was selected because Solita Ltd[2], a consultancy company with over 1000 employees, developed the DigiTraffic API for Fintraffic and two authors of this paper work full time at Solita. Therefore, we are familiar with the technologies, application domain, possible business opportunities, and challenges.

Marine traffic is quintessential for Finland: some 80-90% of exported and imported goods are carried by sea [6]. In winter navigation, the changing ice conditions [7] cause relatively frequently accidents [8] which may trigger, for instance, oil spills and delays [9]. The ecosystem related to marine traffic is large as the total number of companies working in the Finnish maritime cluster[3] is almost 3000 [6].

The marine environment of FSWNS is very special and challenging as it consists of shallow and narrow sea lanes, dense and rocky archipelago, and icy conditions as especially

[1] https://www.digitraffic.fi/en/
[2] http://www.solita.fi
[3] https://shipowners.fi/en/maritime-cluster/

the Bay of Bothnia freezes during the wintertime [10]. These conditions do not only require piloting but also ice breaking in winter when ships may even be guided in a convoy or towed behind an ice breaker. The icebreakers assist vessels free of charge to enable fluent foreign trade. However, unexpected delays of even tens of hours are typical[4] due to changing ice conditions which then affect the inter-modal logistic chain [6]. Nowadays, the icebreaker captains try to interpret an ice forecast[5], a wind forecast[6] and satellite images to predict which vessels might get stuck into the ice and need assistance.

B. Opportunities

Novel data science service utilizing AI technologies could provide help for the decision making process for the icebreaker captains, but also several other parties in the ecosystem would gain benefits from better information: ports, shipping companies, cargo forwarding companies, and transport companies [6] – and their customers. Business-critical opportunities include how to predict and control the estimated time of arrival (ETA) or departure more accurately based on ice and weather conditions. Also, these opportunities may include identifying potential future traffic bottlenecks – waiting does not only mean staying on hold and wasting time at the sea but also that the speed of the vessel could have been lower leading to savings if the waiting was anticipated. Respectively, sudden stop, acceleration, or un-optimized route of a large vessel is equally costly. All these accumulate CO_2 emissions. Many manufacturing and assembly companies also depend on predictive just-in-time import and export in their business processes [6]. The logistics are not limited to marine but include storage, road, and rail. Finally, there is an opportunity to enhance safety, such as predict and prevent a collision in a convoy [11]. Future opportunities lie in technological advancements. Automation is increasingly important in the ports and heavily automated ports already exist outside Finland (e.g., Hamburg and Rotterdam). Also, the ships rely more and more on automation and autonomous cargo ships are being developed and tested.

The public authorities have already made several open data APIs available that are published following the aforementioned legislation. For example, the following APIs are relevant and available: 1) Digitraffic (Ministry of Transport and Communications) traffic APIs including marine[7] and rail[8]; 2) Finpilot piloting status (government-owned piloting company)[9]; 3) Finnish Meteorological Institute weather[10]; 4) many others gathered to the Open data webpage[11]

The increasingly popular open data facilitates novel business opportunities to create data science services for end-customer. That is, AI and especially ML can provide different stakeholders with advanced analytics and predictions based on the open data of these open APIs. Business-critical challenges include how to find a paying customer for open data and build a sustainable software ecosystem: The raw data cannot be the product as basically anyone can access it, so the value for a customer must come from, for example, user experience and good analytics in the right place and time. However, rather than focusing only on the challenges in algorithms and technical solutions, business models, or ecosystems, there are also more general software engineering challenges.

[4]https://vayla.fi/en/transport-network/waterways/winter-navigation
[5]http://baltice.org/weather/
[6]https://www.windy.com
[7]https://www.digitraffic.fi/en/marine-traffic/
[8]https://www.digitraffic.fi/en/railway-traffic/
[9]https://pilotonline.fi/traffic-info/api
[10]https://en.ilmatieteenlaitos.fi/open-data-manual
[11]https://www.avoindata.fi/en

III. Software Engineering Challenges

In this section, we elaborate software engineering related challenges that we have encountered whilst considering different data science service opportunities based on intelligence built on top of open data and APIs for the maritime cluster.

1) Relevant data: REST APIs are today the dominant design in data APIs. It is customary that full data is provided through a REST API with no ability to customize what data is returned. This often leads to fetching a large amount of unnecessary data. For example, Maritime traffic API contains very large JSON messages but sometimes only one piece of data, such as the ETA of a vessel's portcall, is actually needed. That is, the data providers have just opened all data without much considering how data would be best usable. When multiple APIs are utilized and each provides much unnecessary data, this makes the development and running of the system inconvenient and unreliable. There are already be technologies that could potentially alleviate this. For example, GraphQL APIs can drastically reduce the sizes of transferred JSON files [12].

2) Historical data: Historical data, i.e. the data produced and collected over the years, is rarely made available. For example, the Maritime API shows only the current traffic although all data over years has been stored. The data providers might not have considered that someone could find value in the historical data on marine traffic. Alternatively, the historical data can be generalized and provided in a more-coarse grained manner in order not to cause too much load for API, such as only for limited time intervals or limited locations as appears to be for some weather data. However, historical data is required for learning in ML systems. Although a data user can start to store data in order to form a training dataset, storing is slow and inconvenient, and brings forward other challenges, such as licensing below.

3) Licensing: When multiple data sources are involved, different rights become an issue. Unfortunately, often open data and API licenses are even more difficult to manage than those of open source software. For example, common software licenses, such as GPL, BSD or MIT, or content licenses, such as Creative Commons (CC), are not necessarily used for data. Rather, open data providers create their own licenses or do not explicitly mark any license. The licenses can be even hard to find in the API specifications, such as for the pilot data above. Moreover, when data is collected from different sources, it may

be difficult to assess how different licenses are compatible and how the new combined data or solution inferred from data can be licensed and commercialized.

4) Runtime quality: A data science service based on multiple sources that need to be accessed near real-time, emphasizes different runtime quality characteristics, such as reliability and availability. With governmental open data, sudden changes to the open data policies are not perhaps as likely as with other organizations. However, there are no guarantees for dependability or service level agreements (SLAs) at least clearly stated in the data sources. The benefit of data science services often lies in near real-time inference and a discontinuity in source data APIs will immediately affect usefulness. In the worst case, the results can be incorrect rather than unavailable.

5) API Evolution: APIs evolve over time, and the changes often break the client developers' code (e.g. [13]). Generally, the most common API breaking changes are due to refactorings [14]. With open data APIs, the changes have the potential to affect several API users and end-customers – likewise when multiple APIs are used any of them can change and break the solution. A specific problem in ML-based solutions is their lack of fault tolerance if something changes: an ML system can continue its operations and produce incorrect, drifted results if some of its data sources have changed. Identifying the most likely changes in open APIs and preparing for them, e.g., by a means of fault tolerance, can help to mitigate potential API evolution problems. In addition, API evolution can also lead to changes in licensing and technical implementation. This in turn adds yet another layer of complexity in the development process.

IV. A need for governance models

One unifying factor with all the challenges identified in the previous section is that they are all related, to a certain degree, to the governance of the provided APIs and data. *API governance* is defined as *"a task mainly applied inside an organization, typically aiming at achieving a certain harmonization of APIs in terms of their non-functional properties, best-practices-support, documentation quality or rule compliance in general"* [15]. In [16], API governance is seen encompassing a wide range of activities *"starting with the API proposal all the way to its adoption, through requirements gathering, build and deploy, and operations during general availability"*. Data governance, in turn, is associated with decisions regarding the data, i.e., *"data governance refers to who holds the decision rights and is held accountable for an organization's decision-making about its data assets"* [17]. API governance encompasses practices that need to be designed and executed to overcome the challenges of building intelligent data science services on top of open data APIs. As a summary, following aspects needs to be considered when designing API governance [18], [19], [20]:

1) Change Control: When API changes are required, the effects of the change should be predictable and implemented in a uniform, consistent way. If changes need to be rolled back, the return to previous functionality should also be consistent, complete, and managed. This requires the development of efficient and automated change-impact analysis techniques that can determine the potential effects of a proposed change.

2) Impact of Changes: As APIs are created in the context of business, the impact of changes in API should be carefully evaluated. Stakeholders of an API, such as consumers and business owners, should be informed of changes and the possible impact of those.

3) Policy Specification and Analysis: Access control policies, their analysis, and application should be considered to only allow authorized clients to access resources. The accessibility to an API should also consider the business context.

4) Consistent Policy Implementation: Policies that control the use of assets through API should be implemented independently of the technologies that are used to implement the assets. Decoupling API from asset implementation allows for API integrity to be kept changes to one do not influence the other.

5) Life-cycle Alignment: The governance process should be involved in all the duration of the API life-cycle. A governance process should exist for the development, deployment, monitoring, and deprecation of an API.

6) API Integrity: API should be able to interface on a newer version of the platform without conflicts and without effort. When planning new features, existing API should not require extensive refactoring, and backward compatibility should be ensured over a period of time.

7) Monitoring and Auditing: API governance must incorporate a unified method of monitoring and auditing API activity.

Table I summarizes the cross-mapping of API governance aspects and recognized challenges, which helps to understand how API governance could be designed in order to tackle the recognized challenges. To summarize, to guarantee that only *relevant data* is published through API, there is a need to understand the development life-cycle, manage change and understand how changes impact data consumers. Moreover, creating an API that provides *historical data* requires change control as the data structure behind the API can change, but API still needs to provide the same data, whereas *licensing* requires policies to be designed and implemented to manage

TABLE I
RELATING API GOVERNANCE ASPECTS AND RECOGNIZED CHALLENGES.

API governance aspect \ challenge	Relevant data	Historical data	Licensing	Runtime quality	Evolution
Change control		X		X	X
Impact of changes	X			X	X
Policy specification and analysis			X		
Consistent policy implementation			X		
Life-cycle alignment	X			X	X
API integrity			X	X	X
Monitoring and auditing		X		X	X

access to API. Also, API integrity needs to be considered with licensing to manage the compatibility and licenses of different API versions. *Runtime quality* requires quality from both data and the API. API integrity, change control, and acknowledging the impact of changes are major factors to ensure runtime quality. Runtime quality can also be improved by API monitoring and auditing. As the need for runtime quality exists during the whole lifespan of an API, life-cycle alignment is required. Finally, successful *evolution* of API requires managing changes from technical and business perspectives, making sure API integrity exists, aligning API life-cycle, and monitoring and auditing the API.

Moreover, technical API governance and data governance share some responsibilities when it comes to data quality management. As traditional APIs, such as REST and SOAP, are usually built on a separate layer that is not directly connected with data [21], most of the data quality dimensions, e.g. completeness, interpretability, accessibility, and representational consistency are shared responsibilities between data and API governance. In addition, as API governance also contains technical aspects, it needs to be considered as part of the IT governance as well.

API governance can be seen as part of the broader practise of API management, that is described as: *"An activity that enables organizations to design, publish and deploy their APIs for (external) developers to consume. API Management capabilities such as controlling API lifecycles, access and authentication to APIs, monitoring, throttling, and analyzing API usage, as well as providing security and documentation are often implemented through an integrated platform, which is supported by an API gateway."* [22]

API governance can also be seen as a combination of data and IT governance, and as a part of broader API management. There is a need for a governance model that describes the structure of how API governance should be designed and executed. This structure should take into account the different aspects of API governance but enable organizational flexibility that exists because of the heterogeneous nature of business domains.

V. Conclusions

Open data and APIs are increasingly important and available in the future. Many different business opportunities can arise based on data science services, especially relying on machine learning built on top of open data and APIs. However, besides finding a viable business model and algorithms, there are several software engineering challenges when combining open data from several different open APIs for dependable data science services. We outlined our firsthand experiences about the challenges that we have encountered whilst working in the domain of maritime traffic and its open data and APIs. The challenges are not all domain-specific but pertain to data science services built on top of open data and APIs. As a solution to challenges, we discussed how better governance practices at open data and API providers could alleviate these challenges for those who design and operate data science services. We aim to carry out more research on data sciences services built on open data and APIs to gain experiences of fully exploiting the potential in a fully dependable manner. Especially, we are interested in different governance practices in the entire value network.

References

[1] C. González-Mora, D. Tomás, I. Garrigós, J. J. Zubcoff, and J.-N. Mazón, "Model-driven development of web APIs to access integrated tabular open data," *IEEE Access*, vol. 8, pp. 202 669–202 686, 2020.

[2] Open data and reuse of public-sector information. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/LSU/?uri=CELEX%3A32019L1024

[3] S. Martin, M. Foulonneau, S. Turki, and M. Ihadjadene, "Risk analysis to overcome barriers to open data," *Electronic Journal of e-Government*, vol. 11, no. 1, p. 348, 2013.

[4] A. Abelló Gamazo, C. P. Ayala Martínez, C. Farré Tost, C. Gómez Seoane, M. Oriol Hilari, and Ó. Romero Moral, "A data-driven approach to improve the process of data-intensive API creation and evolution," in *International Conference on Advanced Information Systems Engineering (CAiSE-Forum-DC)*, 2017, pp. 1–8.

[5] M. Bergström and P. Kujala, "Simulation-based assessment of the operational performance of the Finnish–Swedish Winter Navigation System," *Applied Sciences*, vol. 10, no. 19, p. 6747, 2020.

[6] U. Tapaninen, *Maritime Transport: Shipping Logistics and Operations*. Kogan Page Publishers, 2020.

[7] O. A. V. Banda, F. Goerlandt, J. Montewka, and P. Kujala, "A risk analysis of winter navigation in Finnish sea areas," *Accident Analysis & Prevention*, vol. 79, pp. 100–116, 2015.

[8] F. Goerlandt, H. Goite, O. A. V. Banda, A. Höglund, P. Ahonen-Rainio, and M. Lensu, "An analysis of wintertime navigational accidents in the Northern Baltic Sea," *Safety science*, vol. 92, pp. 66–84, 2017.

[9] O. A. V. Banda, F. Goerlandt, V. Kuzmin, P. Kujala, and J. Montewka, "Risk management model of winter navigation operations," *Marine pollution bulletin*, vol. 108, no. 1-2, pp. 242–262, 2016.

[10] V. Lehtola, J. Montewka, F. Goerlandt, R. Guinness, and M. Lensu, "Finding safe and efficient shipping routes in ice-covered waters: a framework and a model," *Cold regions science and technology*, vol. 165, p. 102795, 2019.

[11] J. Jussila, T. Lehtonen, J. Laitinen, M. Makkonen, and L. Frank, "Visualising maritime vessel open data for better situational awareness in ice conditions," in *Proceedings of the 22nd International Academic Mindtrek Conference*, 2018, pp. 92–99.

[12] G. Brito and M. T. Valente, "REST vs GraphQL: A controlled experiment," in *International Conference on Software Architecture*, 2020.

[13] A. Brito, M. T. Valente, L. Xavier, and A. Hora, "You broke my code: understanding the motivations for breaking changes in APIs," *Empirical Software Engineering*, vol. 25, no. 2, pp. 1458–1492, 2020.

[14] D. Dig and R. Johnson, "How do APIs evolve? a story of refactoring," *Journal of software maintenance and evolution: Research and Practice*, vol. 18, no. 2, pp. 83–107, 2006.

[15] F. Haupt, F. Leymann, and K. Vukojevic-Haupt, "API governance support through the structural analysis of rest APIs," *Computer Science-Research and Development*, vol. 33, no. 3-4, pp. 291–303, 2018.

[16] B. De, "API governance," in *API Management*. Springer, 2017, pp. 179–188.

[17] V. Khatri and C. V. Brown, "Designing data governance," *Communications of the ACM*, vol. 53, no. 1, pp. 148–152, 2010.

[18] J. Horkoff, J. Lindman, I. Hammouda, and E. Knauss, "Strategic API analysis and planning: APIS technical report," *arXiv preprint arXiv:1911.01235*, 2019.

[19] C. Krintz, H. Jayathilaka, S. Dimopoulos, A. Pucher, R. Wolski, and T. Bultan, "Developing systems for API governance," *figshare*, p. 790746, 2013.

[20] E. Lourenço Marcos and R. Puccinelli de Oliveira, "A framework for guidance of API governance: A design science approach," 2019.

[21] A. Soni and V. Ranga, "API features individualizing of web services: Rest and soap." *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, pp. 664–671, 2019.

[22] M. Mathijssen, M. Overeem, and S. Jansen, "Identification of practices and capabilities in API management: A systematic literature review," *arXiv preprint arXiv:2006.10481*, 2020.